Interpreting CLIP's Image Representation via Text-Based Decomposition

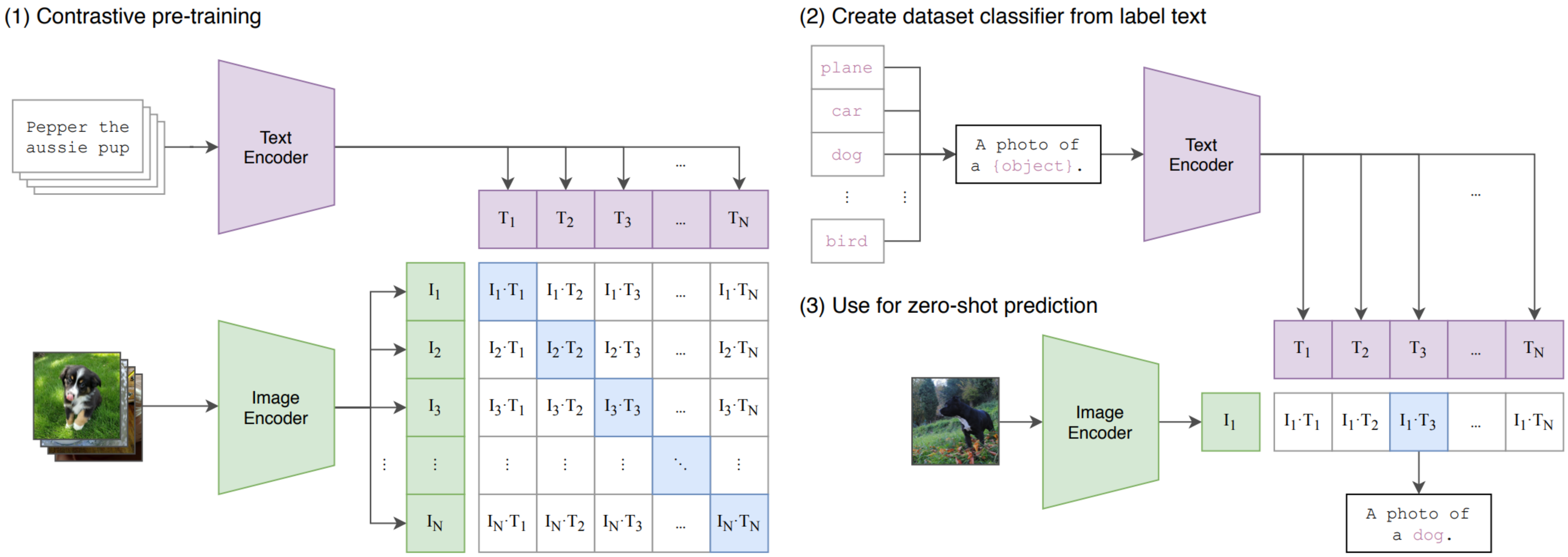
ICLR 2024 Oral

Presenter: Lehong Wu

2024.01.28

Contents

Background: CLIP



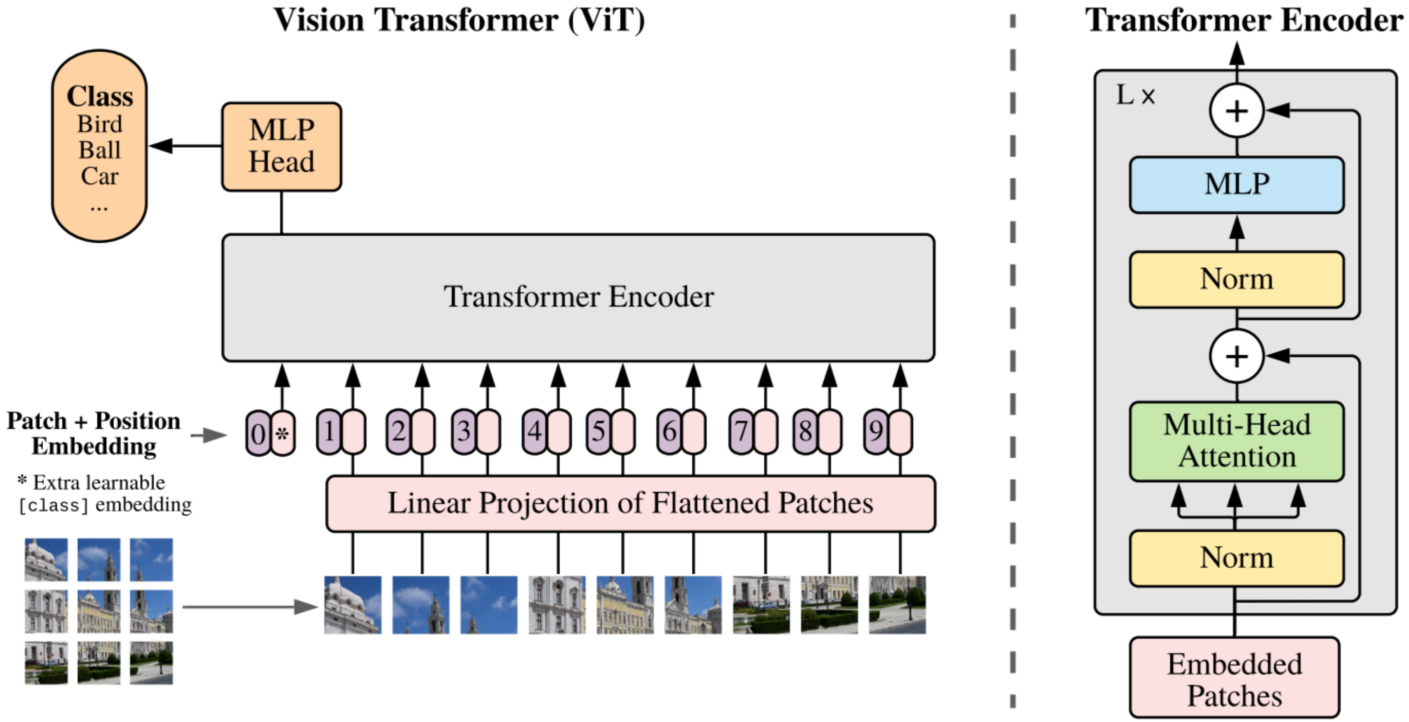
Radford et al. Learning transferable visual models from natural language supervision. https://arxiv.org/abs/2103.00020

Background: CLIP-ViT

- ViT Architecture
  - Multi-head self-attention(MSA)
  - MLP
  - Residual connection
- CLS token as output
- Projected to vision-language space

$$M_{\text{image}}(I) = P\text{ViT}(I)$$

$$P \in \mathbb{R}^{d' \times d}$$

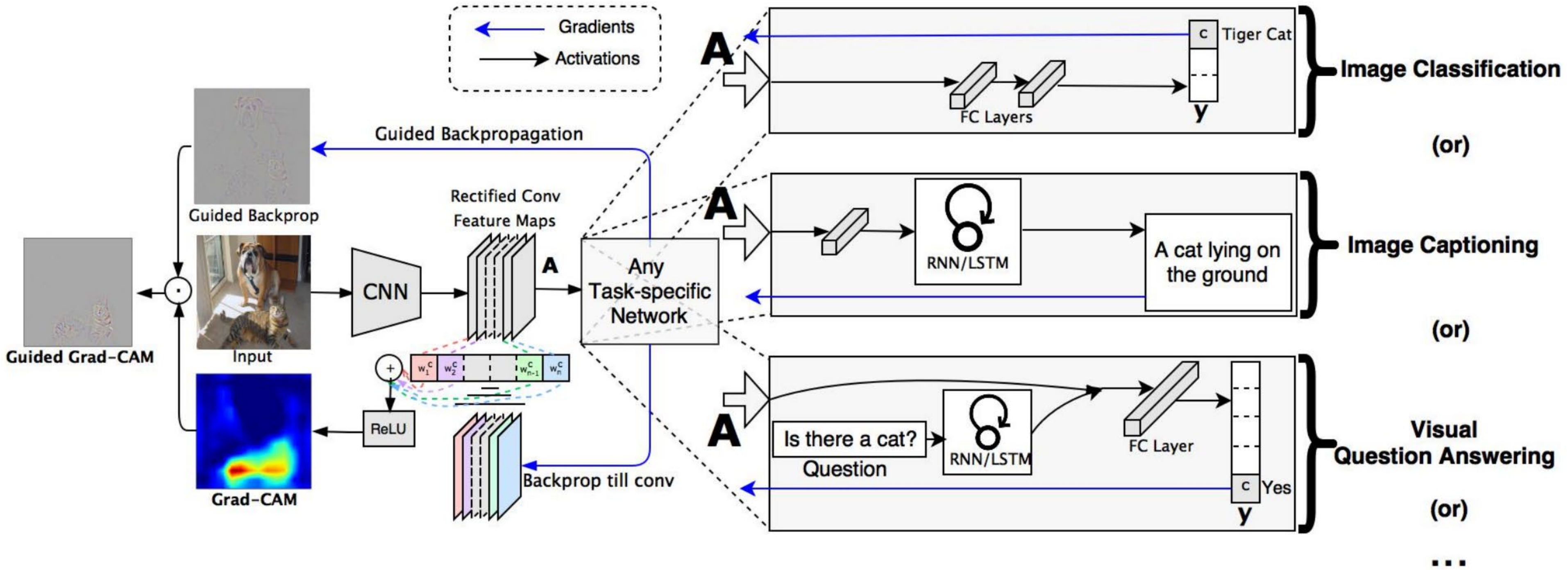


Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. https://arxiv.org/pdf/2010.11929

Background: Heatmap-based Interpretability

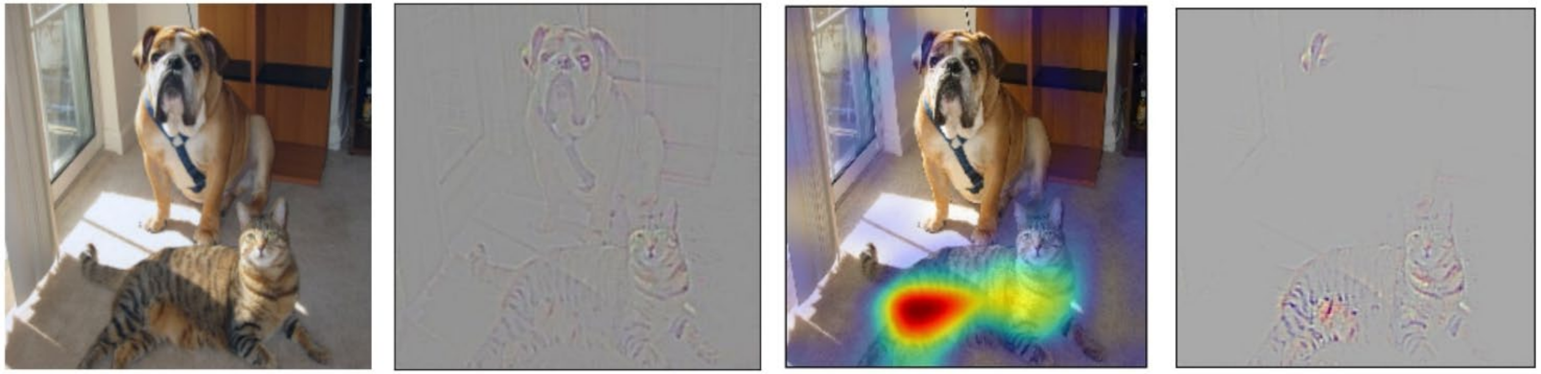
Grad-CAM

- Heuristics: gradient of feature map highlights important regions
- Pixel-level visualization with Guided Backpropagation
- Suppress negative gradient by ReLU



Selvaraju et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. https://arxiv.org/abs/1610.02391

Background: Heatmap-based Interpretability



(a) Original Image (b) Guided Backprop 'Cat' (c) Grad-CAM 'Cat' (d)Guided Grad-CAM 'Cat'

Attribute-based Heatmap Methods

- Define each variable's attribution score to the output
- Define a back propagation rule (e.g. Layer-wise Relevance Propagation)

Heatmap Methods Limitation

- Do not interpret models' intermediate representation
- Only interpret localization-relevant features

Samek et al. Interpreting the Predictions of Complex ML Models by Layer-wise Relevance Propagation. https://arxiv.org/pdf/1611.08191.pdf

Background: Interpretability Summary

- Output Interpretability
  - Heatmap-based (e.g. Grad-CAM, LRP)

    Cons: only interpret localization-relevant features
  - Editing-based

    Use generative models (e.g. StyleGAN) to edit input images

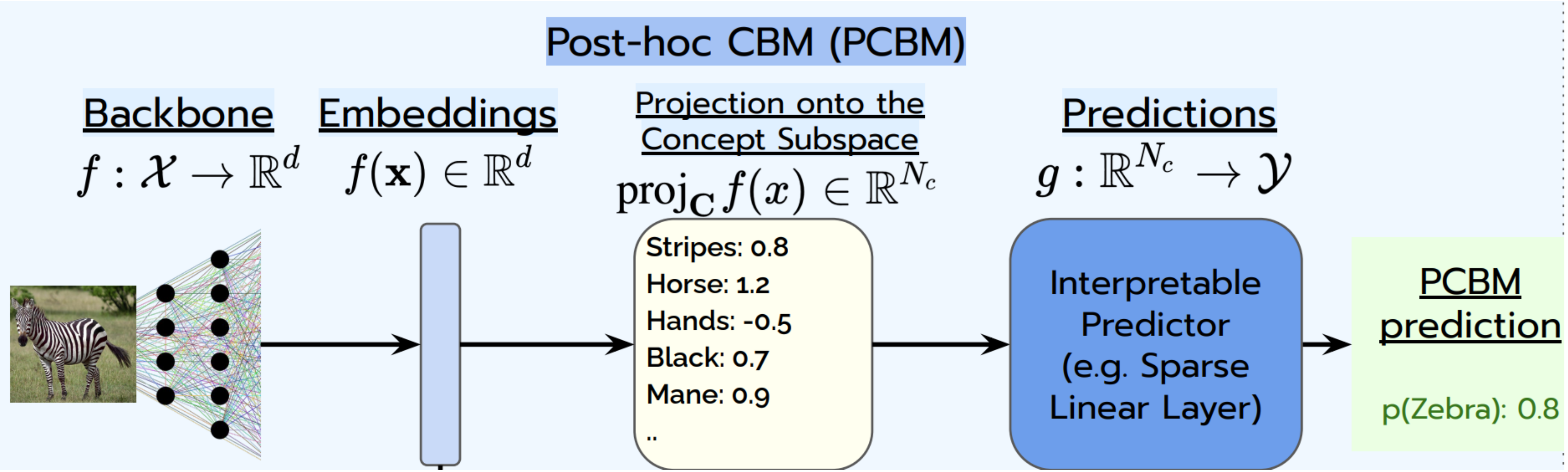
    Cons: rely on generative models
- Representation Interpretability
  - Invert features into image (e.g. Feature Visualization)

    Cons: by optimization; subjectivity
  - Interpret neurons & neuron connections (e.g. Rosetta Neurons)
- Text-based Interpretability
  - Use text to describe images/representations
  - Interpret CLIP representations

Background: Text-based Interpretability

Post-hoc Concept Bottleneck Model

- Map features to a text-based concept space
- Any pre-trained image encoder
- Concept space constructed with a text encoder (e.g. CLIP)

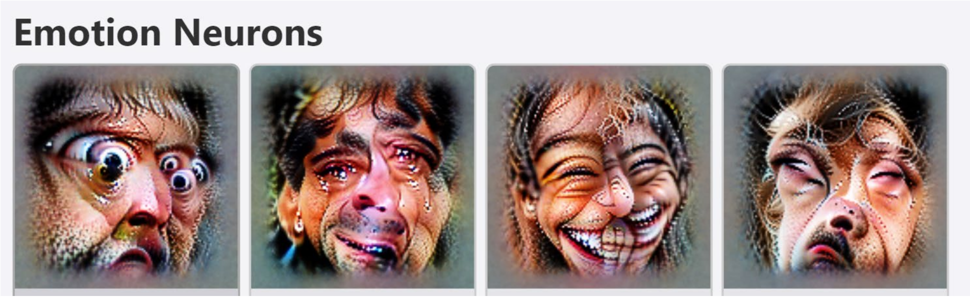


Yuksekgonul et al. Post-hoc concept bottleneck models. https://arxiv.org/abs/2205.15480

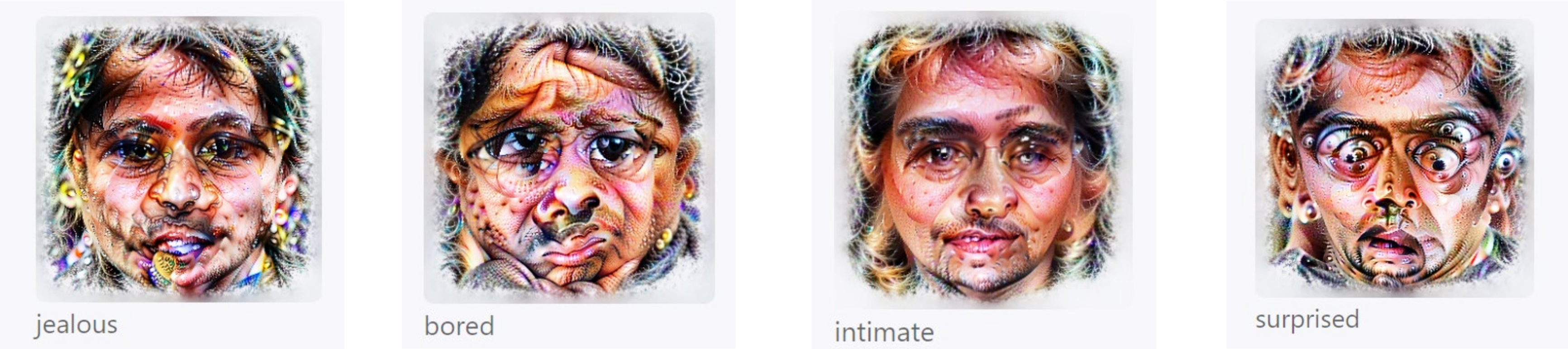
Background: CLIP Interpretability

Feature Visualization of CLIP

- Maximize neuron activation



- Maximize similarity with the given text

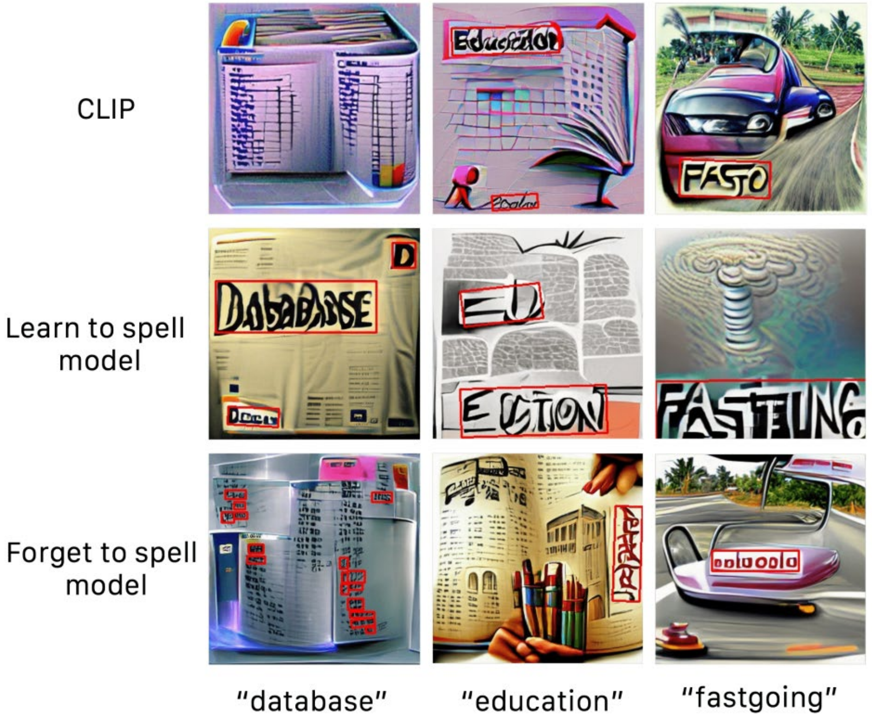


Goh et al. Multimodal neurons in artificial neural networks. https://distill.pub/2021/multimodal-neurons

Background: CLIP Interpretability

Understanding CLIP's Spelling capability

- Train a learn-to-spell and a forget-to-spell model by optimizing opposite objectives



Materzynska et al. Disentangling visual and written concepts in CLIP. https://arxiv.org/abs/2206.07835

Method: Overview

- Decompose CLIP representation into direct contributions of each layer, each attention ***head***, and each ***position*** (image token)
- Understanding ***heads***
  - Technique: label heads with text
  - Application: reduce spurious correlation; property-specific image retrieval
- Understanding ***positions***
  - Technique: heatmap
  - Application: zero-shot segmentation

Method: Decomposition into Layers

- Each module directly updates the residual stream:

$$\hat{Z}^l = \mathsf{MSA}^l(Z^{l-1}) + Z^{l-1}, \quad Z^l = \mathsf{MLP}^l(\hat{Z}^l) + \hat{Z}^l$$

- Decompose the final representation:

$$M_{\text{image}}(I) = P\mathsf{ViT}(I) = P\left[Z^0\right]_{cls} + \underbrace{\sum_{l=1}^{L} P\left[\mathsf{MSA}^l(Z^{l-1})\right]_{cls}}_{\text{MSA terms}} + \underbrace{\sum_{l=1}^{L} P\left[\mathsf{MLP}^l(\hat{Z}^l)\right]_{cls}}_{\text{MLP terms}}$$

- ***Direct*** effects: components in the equation (this paper)

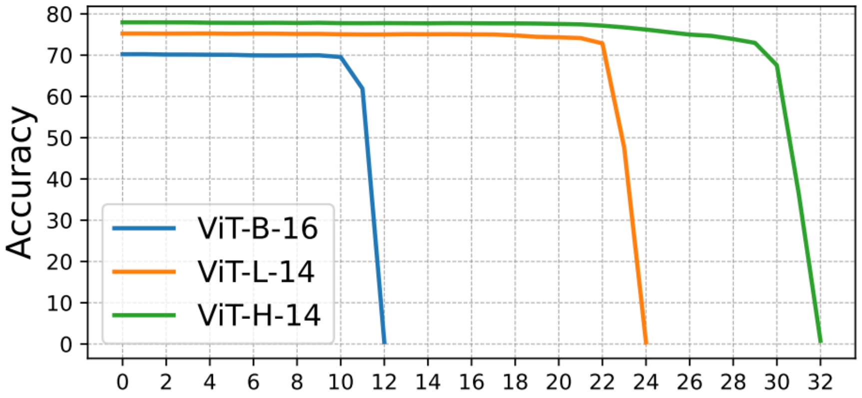
  Indirect effects: influence of early layers on later layers

Experiments: Decomposition into Layers

- Use OpenCLIP trained on LAION-2B
- Mean-ablation: replace ***output*** components with mean values across the dataset in zero-shot classification task

	Base accuracy	+ MLPs ablation
ViT-B-16	70.22	67.04
ViT-L-14	75.25	74.12
ViT-H-14	77.95	76.30

MLP mean-ablation



MSA mean-ablation

- Conclusion
  - MLPs have negligible direct effects
  - Only the last MSAs have significant direct effects (this paper)

Method: Decomposition into Heads & Positions

- One MSA component: $\left[\mathsf{MSA}^l(Z^{l-1})\right]_{cls} = \sum_{h=1}^{H}\sum_{i=0}^{N} x_i^{l,h}, \quad x_i^{l,h} = \alpha_i^{l,h} W_{VO}^{l,h} z_i^{l-1}$
- All MSA components: $\sum_{l=1}^{L} P\left[\mathsf{MSA}^l(T^{l-1})\right]_{cls} = \sum_{l=1}^{L}\sum_{h=1}^{H}\sum_{i=0}^{N} c_{i,l,h}, \quad c_{i,l,h} = P x_i^{l,h}$

where i, l, h represents token, layer, head

- Contract along certain dimension
    - Head contribution $c_{\text{head}}^{l,h} = \sum_{i=0}^{N} c_{i,l,h}$
    - Position(token) contribution $c_{\text{token}}^{i} = \sum_{l=1}^{L}\sum_{h=1}^{H} c_{i,l,h}$

Method: Understanding Heads

北京大学 PEKING UNIVERSITY

- Goal:

  Find descriptions explaining the "principal components" of a head's contribution $c_{\text{head}}^{l,h}$

- Problem definition:

  $\mathcal{T}$: the set of text descriptions to find, size m hyperparam

  $c_1, c_2, \ldots, c_K$: the head's contribution of all images

  $Proj_{\mathcal{T}}$: projection onto the span of text representations in $\mathcal{T}$

- Maximize

$$V_{\text{explained}}(\mathcal{T}) = \frac{1}{K}\sum_{k=1}^{K} \|\operatorname{Proj}_{\mathcal{T}}(c_k - c_{\text{avg}})\|_2^2, \text{ where } c_{\text{avg}} = \frac{1}{K}\sum_{k=1}^{K} c_k.$$

Method: Understanding Heads

- Maximize $V_{\text{explained}}(\mathcal{T}) = \frac{1}{K}\sum_{k=1}^{K} \| \text{Proj}_{\mathcal{T}}(c_k - c_{\text{avg}}) \|_2^2$, where $c_{\text{avg}} = \frac{1}{K}\sum_{k=1}^{K} c_k$.

- Algorithm: TextSpan
  - Step1: initialize a description pool using ChatGPT-3.5
  - Step2: ***greedily*** select the description with highest variance of projection
  - Step3: update all vectors to be orthogonal to the selected text representation
  - Step4: repeat Step2-3 until *m* times

Method: Understanding Heads

Algorithm 1: TEXTSPAN

Input: Head (l, h) contribution $c_{\text{head}}^{l,h}$ for K images stacked as rows in a matrix $C \in \mathbb{R}^{K \times d'}$, a pool of M text descriptions $\{t_i\}_{i=1}^{M}$, their corresponding CLIP text representations $R \in \mathbb{R}^{M \times d'}$ (projected to the head output space), and basis size m

Output: A set of text descriptions $\mathcal{T}$ and projected representations $C' \in \mathbb{R}^{K \times d'}$

Initialization: $C' \leftarrow \mathbf{0}_{K \times d'}, \mathcal{T} \leftarrow \phi$

for i *in* $[1, ..., m]$ **do**

$D \leftarrow RC^T$

$j^* \leftarrow \arg\max_{j=1}^{M} \operatorname{Var}(D[j])$ → Greedy selection

$\mathcal{T} \leftarrow \mathcal{T} \cup \{t_{j^*}\}$

for k *in* $[1, ..., K]$ **do**

$C'[k] \leftarrow C'[k] + \frac{\langle C[k], R[j^*] \rangle}{||R[j^*]||^2} R[j^*]$

$C[k] \leftarrow C[k] - \frac{\langle C[k], R[j^*] \rangle}{||R[j^*]||^2} R[j^*]$ → Update contributions

for k *in* $[1, ..., M]$ **do**

$R[k] \leftarrow R[k] - \frac{\langle R[k], R[j^*] \rangle}{||R[j^*]||^2} R[j^*]$ → Update text representations

Experiments: Understanding Heads

L21.H11 (“Geo-locations”)	**L23.H10** (“Counting”)
Photo captured in the Arizona desert	Image with six subjects
Picture taken in Alberta, Canada	Image with four people
Photo taken in Rio de Janeiro, Brazil	An image of the number 3
Picture taken in Cyprus	An image of the number 10
Photo taken in Seoul, South Korea	The number fifteen
L22.H11 (“Colors”)	**L22.H6** (“Animals”)
A charcoal gray color	Curious wildlife
Sepia-toned photograph	Majestic soaring birds
Minimalist white backdrop	An image with dogs
High-contrast black and white	Image with a dragonfly
Image with a red color	An image with cats
L23.H12 (“Textures”)	**L22.H1** (“Shapes”)
Artwork with pointillism technique	A semicircular arch
Artwork with woven basket design	An isosceles triangle
Artwork featuring barcode arrangement	An oval
Image with houndstooth patterns	Rectangular object
Image with quilted fabric patterns	A sphere

Layer 20, Head 12
Photo with grainy, old film effect
Detailed illustration
Serene beach sunset
An image of the number 10
An image of the number 5

Layer 21, Head 5
Inquisitive facial expression
Artwork featuring typographic patterns
A photograph of a big object
Reflective landscape
Burst of motion

Heads without clear roles.

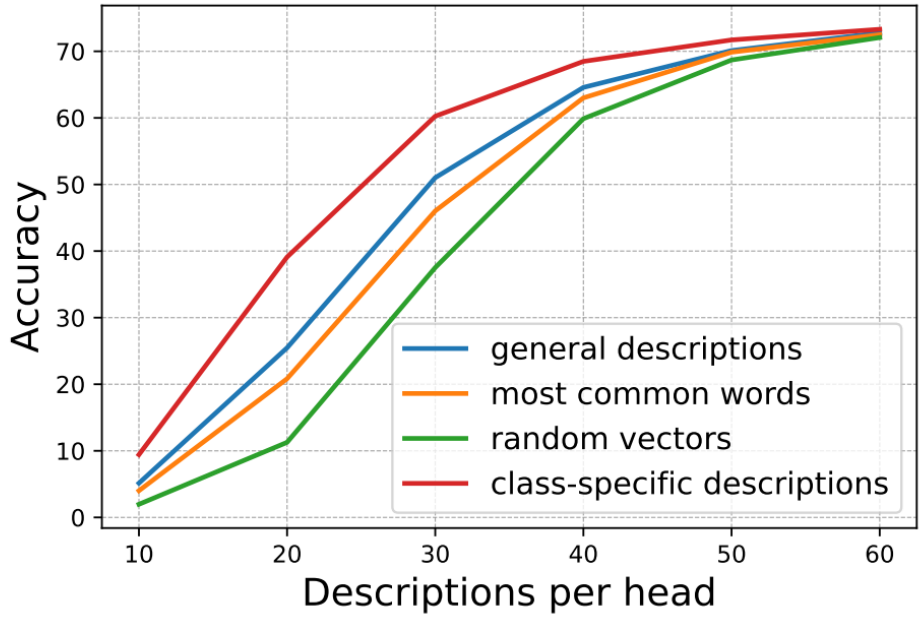
Top-5 results of TextSpan applied to the last 4 layers of CLIP-ViT-L.

The roles of heads in brackets are annotated ***by human***.

Experiments: Understanding Heads

Ablation study

- Project representation to the TextSpan bases
- Classification as evaluation
- Dataset: ImageNet
- Results
  - ChatGPT descriptions is better than common words in English & random vectors
  - Larger basis size m is better (60 enough for 768 dims)
  - Class-specific descriptions (28k) better than general descriptions (3.5k)



Experiments: Understanding Heads

Application 1: Reducing spurious cues for classification

- Dataset: Waterbirds
- Setting: Zero-shot classification
- *Manually* mean-ablated the heads relevant to "location", according to TextSpan results

	base	top random	ours
ViT-B-16	45.6	52.3	**57.5**
ViT-L-14	47.7	57.7	**72.9**
ViT-H-14	37.2	37.0	**43.3**

Comparison with CLIP and random ablation.

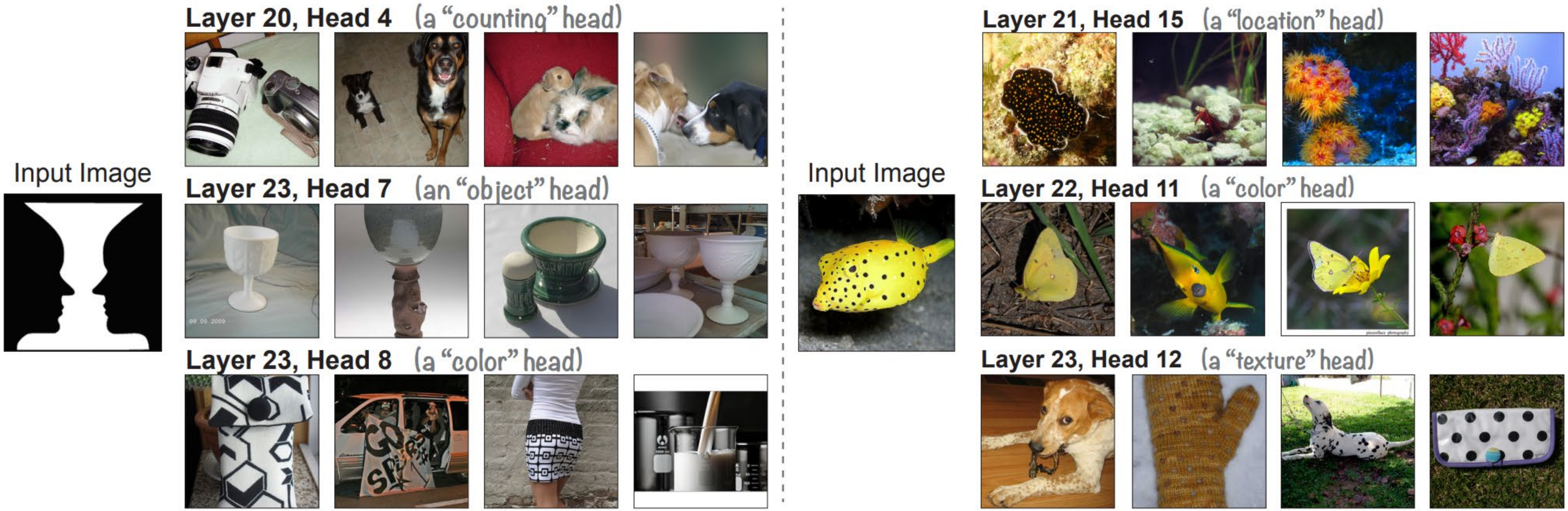
	water background	land background
waterbird class	92.1 (**93.1**)	**77.8** (66.2)
landbird class	**72.9** (47.7)	**94.9** (94.8)

Detailed comparison with CLIP.

Experiments: Understanding Heads

Application 2: Property-based image retrieval

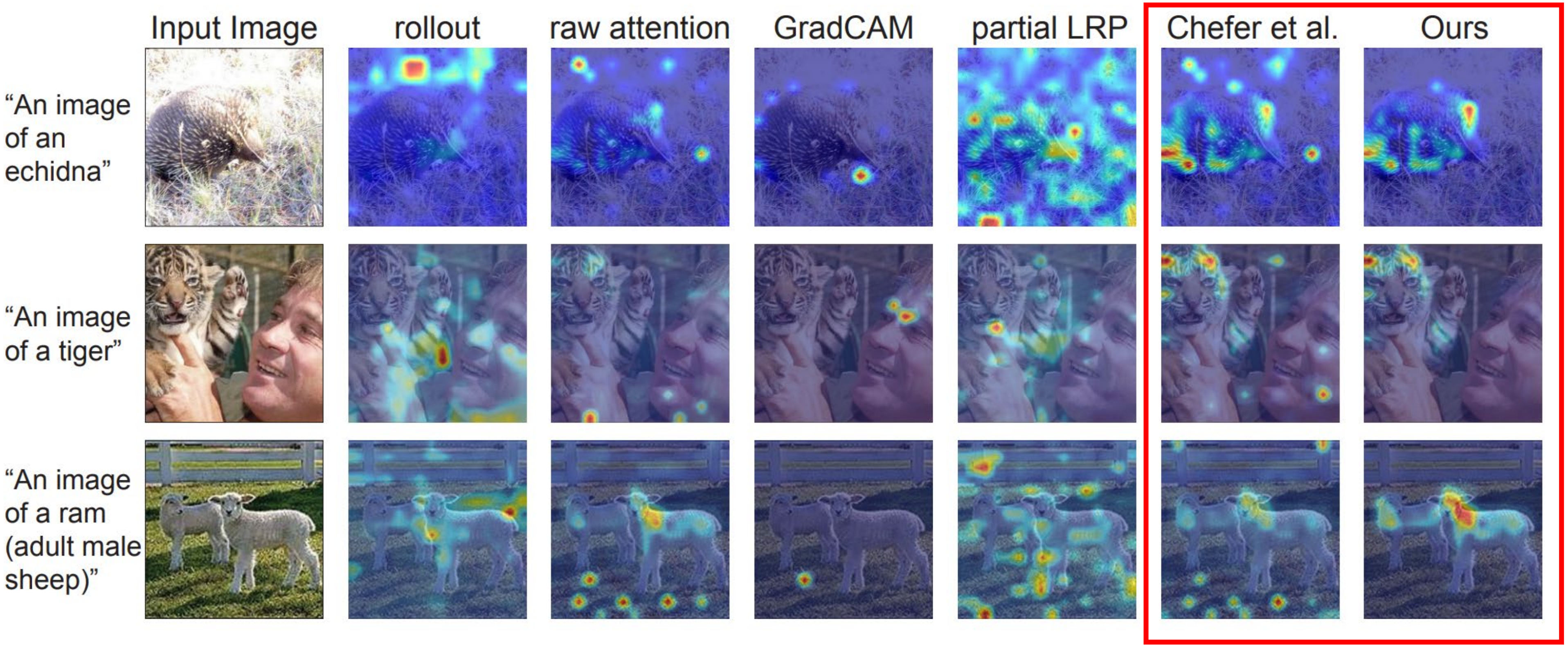
- Given a base image, retrieve its nearest neighbors for a certain head (by computing cosine similarity)



Experiments: Understanding Positions

Same as evaluating heatmap-based methods

- Heatmap visualization by calculation similarity between tokens & text representations
- Highlighted regions are more aligned with the text



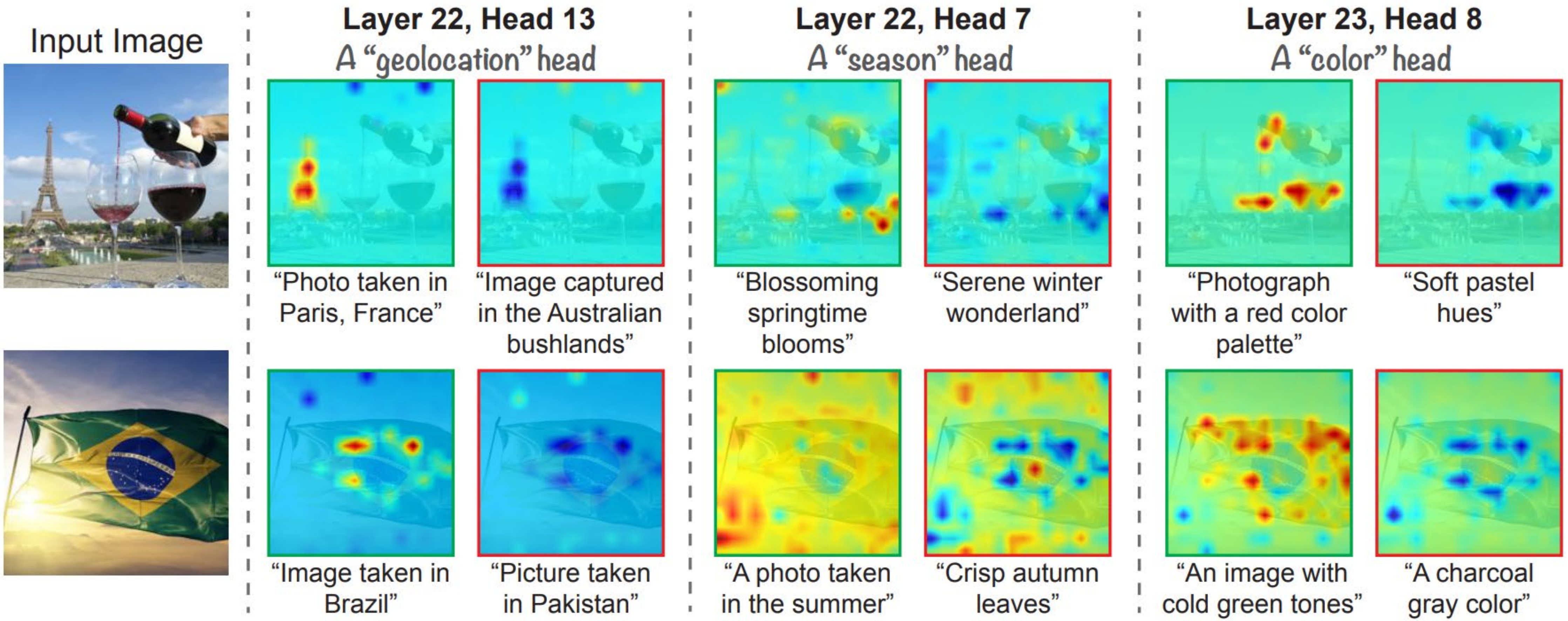
Experiments: Understanding Positions

Zero-shot segmentation

- Method: binarize heatmap (with a threshold) to obtain a foreground / background segmentation
- Dataset: ImageNet-Segmentation (4,276 images from 445 categories)

	Pixel Acc. ↑	mIoU ↑	mAP ↑
LRP (Binder et al., 2016)	52.81	33.57	54.37
partial-LRP (Voita et al., 2019)	61.49	40.71	72.29
rollout (Abnar & Zuidema, 2020)	60.63	40.64	74.47
raw attention	65.67	43.83	76.05
GradCAM Selvaraju et al. (2017)	70.27	44.50	70.30
Chefer et al. (2021)	69.21	47.47	78.29
Ours	**75.21**	**54.50**	**81.61**

Experiments: Understanding Heads & Positions



Green / Red border heatmaps correspond to the descriptions most / least similar to $c_{head}^{l,h}$ among TextSpan outputs.

Conclusion

- Interpreting CLIP image encoder by annotating heads with texts
- Application: removing spurious correlation, image retrieval
- Limitations and discussion:
  - Indirect effects?
  - Not all heads have clear roles
  - Heads are annotated manually
  - Analysis on text encoder and other architectures?

Thanks for listening!

Presenter: Lehong Wu
2024.01.28